Amendments to the Claims:

Claim 1. (Canceled)

2. (Currently amended) The method of claim 3 ~~1~~, wherein the step of computing the global statistics further comprises the steps of estimating the global noise standard deviation σ to generate the global statistics.

3. (Currently Amended) ~~The method of claim 1, wherein the step of computing the local statistics for each pixel further includes the steps of:~~ A method for reducing noise in a digital image formed from a plurality of pixels including a given pixel, the method comprising the steps of:

computing global statistics from the image;

computing local statistics for the given pixel;

configuring a local filter using the local and global statistics;

filtering the given pixel using the local filter to reduce image noise;

wherein the step of computing the local statistics for the given pixel further includes the steps of:

selecting a window containing the given ~~said~~ pixel and a plurality of neighboring pixels;

computing a ~~the~~ 2-D local variance of the given ~~said~~ pixel based on information related to the pixels in the window;

computing a plurality of ~~the~~ 1-D local variances along multiple directions through the given ~~said~~ pixel in ~~within~~ the window; and

detecting a ~~the~~ local edge direction by selecting one of the direction with the smallest 1-D local variance.

4. (Currently amended) The method of claim 3 ~~1~~, wherein the step of computing the local statistics for the given pixel, ~~each pixel~~ further includes the steps of:

selecting a window containing the given ~~said~~ pixel and a plurality of neighboring pixels;

computing the 2-D local variance σ_0^2 of the given ~~said~~ pixel based on information related to the pixels in the window;

computing the 1-D local variances σ_1^2, σ_2^2, σ_3^2, and σ_4^2 along the horizontal (L_1), vertical (L_2), diagonal from upper left to lower right (L_3), and diagonal from upper right to lower left (L_4) directions through the given ~~said~~ pixel, respectively, in ~~within~~ the window; and

detecting the local edge direction by selecting ~~one of~~ the direction ~~directions~~ with the smallest 1-D local variance.

5. (Currently amended) The method of claim 3 ~~1~~, wherein the step of configuring the local filter ~~for each pixel~~ using the local and global statistics further includes the steps of:

selecting the detected local edge direction L as the direction of the local filter;

for the detected local edge direction L computing the 1-D filter strength as a function of the square root of the local variance and the global noise standard deviation;

computing the 2-D filter strength as a function of the local variance and the global noise standard deviation; and

configuring the local filter for the detected local edge direction L based on the 1-D and 2-D filter strengths.

6. (Currently amended) The method of claim 5, wherein the step of configuring the local filter ~~for each pixel~~ using the local and global statistics further includes the steps of:

selecting the detected local edge direction L_k (k = 1, 2, 3, or 4) as the direction of the local filter;

for the detected local edge direction L_k computing the 1-D filter strength $\alpha_k = \min(2\sigma, \max(3\sigma - \sigma_k, 0))/(2\sigma)$;

computing the 2-D filter strength $\alpha_0 = \min(2\sigma, \max(3\sigma - \sigma_0, 0))/(2\sigma)$; and

configuring the local filter f_k for the detected local edge direction L_k according to the following conditions, wherein α_k is the filter strength along the local edge direction L_k [k = 0 (non-edge), 1 (horizontal), 2 (vertical), 3 (upper left to lower right, 4 (upper right to lower left)]:

(i) if the detected direction is L_1, then f_1 is configured as a 2-D local filter for horizontal direction, wherein:

$$f_1 = \frac{1}{9}\begin{bmatrix} \alpha_0 & \alpha_0 & \alpha_0 \\ \alpha_0 + 3\alpha_1(1-\alpha_0) & \alpha_0 + 3(3-2\alpha_1)(1-\alpha_0) & \alpha_0 + 3\alpha_1(1-\alpha_0) \\ \alpha_0 & \alpha_0 & \alpha_0 \end{bmatrix};$$

(ii) if the detected direction is L_2, then f_2 is configured as a 2-D local filter for vertical direction, wherein:

$$f_2 = \frac{1}{9}\begin{bmatrix} \alpha_0 & \alpha_0 + 3\alpha_2(1-\alpha_0) & \alpha_0 \\ \alpha_0 & \alpha_0 + 3(3-2\alpha_2)(1-\alpha_0) & \alpha_0 \\ \alpha_0 & \alpha_0 + 3\alpha_2(1-\alpha_0) & \alpha_0 \end{bmatrix};$$

(iii) if the detected direction is L_3, then f_3 is configured as a 2-D local filter for the diagonal direction from upper left to lower right, wherein:

$$f_3 = \frac{1}{9}\begin{bmatrix} \alpha_0 + 3\alpha_3(1-\alpha_0) & \alpha_0 & \alpha_0 \\ \alpha_0 & \alpha_0 + 3(3-2\alpha_3)(1-\alpha_0) & \alpha_0 \\ \alpha_0 & \alpha_0 & \alpha_0 + 3\alpha_3(1-\alpha_0) \end{bmatrix}; \text{ and}$$

(iv) if the detected direction is L_4, then f_4 is configured as a 2-D local filter for the diagonal direction from upper right to lower left, wherein:

$$f_4 = \frac{1}{9}\begin{bmatrix} \alpha_0 & \alpha_0 & \alpha_0 + 3\alpha_4(1-\alpha_0) \\ \alpha_0 & \alpha_0 + 3(3-2\alpha_4)(1-\alpha_0) & \alpha_0 \\ \alpha_0 + 3\alpha_4(1-\alpha_0) & \alpha_0 & \alpha_0 \end{bmatrix}.$$

7. (Currently amended) The method of claim 2, wherein the steps of estimating the global noise standard deviation σ further includes the steps of:

dividing the input image into overlapping or non-overlapping blocks;

computing a ~~the~~ mean and a ~~the~~ standard deviation for each block;

finding the smallest standard deviation d_0 _ and its corresponding mean _ m_0;

detecting block saturation due to noise;

compensating for the smallest standard deviation d_0 to generate a compensated smallest standard deviation $\tilde{d}_0$;

selecting the block standard deviations d_n _ that are within a range of the compensated smallest standard deviation $\tilde{d}_0$ _; and

averaging the selected block standard deviations d_n to generate an estimate of the global noise standard deviation σ.

8. (Currently amended) The method of claim 7, wherein the block size is 7 _ $\times$ _ 7 or 5×9 _ pixels.

9. (Currently amended) The method of claim 7, wherein the steps of detecting saturation and compensating the smallest standard deviation further include the steps of ~~determining the following~~:

defining ~~is~~ an upper pixel value limit *UL*, a lower pixel value limit *LL*, and mid value *M* between *UL* and *LL*,

wherein if the mean m_0 is less than the mid <u>value</u> ~~range~~ *M*, and the smallest standard deviation is greater than <u>a</u> ~~the~~ difference between the mean m_0 _ and the lower limit *LL*, then saturation has occurred at the lower limit *LL*, and the smallest

standard deviation d_0 is compensated by adding thereto a compensation term that is a function of the smallest standard deviation d_0 and said difference between the mean m_0 and the lower limit *LL*, to generate the compensated smallest standard deviation $\tilde{d}_0$;

else if the mid value ~~range~~ *M* is less than the mean m_0, and the smallest standard deviation d_0 is greater than the difference between the upper limit *UL* and the mean m_0, then saturation has occurred at the upper limit *UL*, and the smallest standard deviation d_0 is compensated by adding thereto a compensation term that is a function of the smallest standard deviation d_0 and the said difference between the upper limit *UL* and the mean m_0, to generate the compensated smallest standard deviation $\tilde{d}_0$;

otherwise, no saturation has occurred, wherein $\tilde{d}_0 = d_0$.

10. (Currently amended) The method of claim 7, wherein the steps of detecting saturation and compensating the smallest standard deviation further include the steps of determining the following:

where *UL* is an upper pixel value limit, *LL* is a lower pixel value limit, *M* is a mid pixel value, and $UL < M < LL$, if the mean $m_0 < M$ and the smallest standard deviation $d_0 > m_0 - LL$, then saturation has occurred at the lower limit *LL*, wherein d_0 is compensated as $\tilde{d}_0 = d_0 + K \cdot (d_0 - (m_0 - LL))$, such that K is a compensation factor;

else if the mean $m_0 \geq M$ and the smallest standard deviation $d_0 > UL - m_0$, then saturation has occurred at the upper limit *UL*, wherein d_0 is compensated as $\widetilde{d}_0 = d_0 + K \cdot (d_0 - (UL - m_0))$;

otherwise, no saturation has occurred, wherein $\widetilde{d}_0 = d_0$.

11. (Previously presented) The method of claim 10, wherein $LL = 0$, $UL = 255$, and $M = 128$.

12. (Previously presented) The method of claim 7, wherein the step of selecting the block standard deviations further includes the steps of selecting the block standard deviation d_n for averaging if $| d_n - \widetilde{d}_0 | < \max(\widetilde{d}_0, 1)$.

Claim 13. (Canceled)

14. (Currently amended) The system of claim 15 ~~13~~, wherein the global statistics module estimates a ~~the~~ global noise standard deviation σ to generate the global statistics.

15. (Currently amended) ~~The system of claim 13,~~ A noise reduction system for reducing noise in a digital image comprising pixels, the system comprising:

a global statistics module that computes global statistics from the image;

a local statistics module that computes local statistics for each of a plurality of image pixels including a given pixel;

a filter configuration module that uses the local and global statistics for the given pixel to configure a local filter for filtering the given pixel;

the local filter as configured by the filter configuration module, being adapted for filtering the given pixel to reduce image noise, wherein the local statistics module computes the local statistics for the given ~~each~~ pixel by:

selecting a window containing the given ~~said~~ pixel and a plurality of neighboring pixels;

computing a ~~the~~ 2-D local variance of said pixel based on information related to the pixels in the window;

computing a plurality of ~~the~~ 1-D local variances along multiple directions each defined by a pair of the pixels ~~through said pixel~~ in ~~within~~ the window; and

detecting a ~~the~~ local edge direction for the given pixel by selecting ~~one of~~ the direction with the smallest 1-D local variance.

16. (Currently amended) The system of claim 15 ~~13~~, wherein the local statistics module computes the local statistics for each pixel by:

selecting a window containing said pixel and a plurality of neighboring pixels;

computing the 2-D local variance σ_0^2 of the given ~~said~~ pixel based on information related to the pixels in the window;

computing the 1-D local variances σ_1^2, σ_2^2, σ_3^2, and σ_4^2 along the horizontal (L_1), vertical (L_2), diagonal from upper left to lower right (L_3), and diagonal

from upper right to lower left (L_4) directions through the given ~~said~~ pixel, respectively, in ~~within~~ the window; and

detecting the local edge direction by selecting ~~one of~~ the direction ~~directions~~ with the smallest 1-D local variance.

17. (Currently amended) The system of claim 15 ~~13~~, wherein the filter configuration module configures the local filter for each pixel using the local and global statistics by:

selecting a ~~the~~ detected local edge direction L as the direction of the local filter;

for the detected local edge direction L computing a ~~the~~ 1-D filter strength as a function of the square root of the local variance and the global noise standard deviation;

computing a ~~the~~ 2-D filter strength as a function of a ~~the~~ local variance and a ~~the~~ global noise standard deviation; and

configuring the local filter for the detected local edge direction L based on the 1-D and 2-D filter strengths.

18. (Currently amended) The system of claim 17, wherein the filter configuration module configures ~~configured~~ the local filter for each pixel using the local and global statistics, wherein α_k is the filter strength and σ_k is the global noise standard deviation for edge directions L_k [k = 0 (non-edge), 1 (horizontal), 2 (vertical), 3 (upper left to lower right), 4 (upper right to lower left)] by:

selecting the detected local edge direction L_k (k = 1, 2, 3, or 4) as the direction of the local filter;

for the detected local edge direction L_k,

computing the 1-D filter strength $\alpha_k = \min(2\sigma, \max(3\sigma - \sigma_k, 0))/(2\sigma)$,

and [[;]] ~~computing~~ the 2-D filter strength

$\alpha_0 = \min(2\sigma, \max(3\sigma - \sigma_0, 0))/(2\sigma)$; and

configuring the local filter f_k for the detected local edge direction L_k according to the following conditions:

(i) if the detected direction is L_1, then f_1 is configured as a 2-D local filter for horizontal direction, wherein:

$$f_1 = \frac{1}{9}\begin{bmatrix} \alpha_0 & \alpha_0 & \alpha_0 \\ \alpha_0 + 3\alpha_1(1-\alpha_0) & \alpha_0 + 3(3-2\alpha_1)(1-\alpha_0) & \alpha_0 + 3\alpha_1(1-\alpha_0) \\ \alpha_0 & \alpha_0 & \alpha_0 \end{bmatrix};$$

(ii) if the detected direction is L_2, then f_2 is configured as a 2-D local filter for vertical direction, wherein:

$$f_2 = \frac{1}{9}\begin{bmatrix} \alpha_0 & \alpha_0 + 3\alpha_2(1-\alpha_0) & \alpha_0 \\ \alpha_0 & \alpha_0 + 3(3-2\alpha_2)(1-\alpha_0) & \alpha_0 \\ \alpha_0 & \alpha_0 + 3\alpha_2(1-\alpha_0) & \alpha_0 \end{bmatrix};$$

(iii) if the detected direction is L_3, then f_3 is configured as a 2-D local filter for the diagonal direction from upper left to lower right, wherein:

$$f_3 = \frac{1}{9}\begin{bmatrix} \alpha_0 + 3\alpha_3(1-\alpha_0) & \alpha_0 & \alpha_0 \\ \alpha_0 & \alpha_0 + 3(3-2\alpha_3)(1-\alpha_0) & \alpha_0 \\ \alpha_0 & \alpha_0 & \alpha_0 + 3\alpha_3(1-\alpha_0) \end{bmatrix}; \text{ and}$$

(iv) if the detected direction is L_4, then f_4 is configured as a 2-D local filter for the diagonal direction from upper right to lower left, wherein:

$$f_4 = \frac{1}{9}\begin{bmatrix} \alpha_0 & \alpha_0 & \alpha_0 + 3\alpha_4(1-\alpha_0) \\ \alpha_0 & \alpha_0 + 3(3-2\alpha_4)(1-\alpha_0) & \alpha_0 \\ \alpha_0 + 3\alpha_4(1-\alpha_0) & \alpha_0 & \alpha_0 \end{bmatrix}.$$

19. (Currently amended) The system of claim 14, wherein the input image comprises a plurality of ~~overlapping or non-overlapping~~ blocks, and wherein the global statistics module further comprises:

a mean and standard deviation module that computes the mean and the standard deviation for each block ;

a minimum finder module that finds the smallest standard deviation d_0 and its corresponding mean m_0;

a saturation detector that detects block saturation due to noise;

a saturation compensator that compensates for the smallest standard deviation d_0 to generate a compensated smallest standard deviation $\tilde{d}_0$; and

a selective averaging module that selects the block standard deviations d_n that are within a range of the compensated smallest standard deviation $\tilde{d}_0$, and averages the selected block standard deviations d_n to generate an estimate of the global noise standard deviation σ.

20. (Currently amended) The system of claim 19, wherein the block size is $7\_\times\_7$ or $5\times 9\_$ pixels.

21. (Currently amended) The system of claim 19, wherein:

an upper pixel value limit is denoted *UL*, a lower pixel value limit is denoted *LL*, and a mid value *M* is between *UL* and *LL*,

wherein the saturation detector determines if the mean m_0 is less than the mid range *M*, and the smallest standard deviation is greater than a ~~the~~ difference between the mean m_0 and the lower limit *LL*, indicating that saturation has occurred at the lower limit *LL*, and if so, the saturation compensator compensates the smallest standard deviation d_0 [[is]] by adding thereto a compensation term that is a function of the smallest standard deviation d_0 and said difference between the mean m_0 and the lower limit *LL*, to generate the compensated smallest standard deviation $\tilde{d}_0$;

else if the saturation detector determines that the mid value ~~range~~ *M* is less than the mean m_0, and the smallest standard deviation d_0 is greater than a ~~the~~ difference between the upper limit *UL* and the mean m_0, indicating saturation has occurred at the upper limit *UL*, the saturation compensator compensates the smallest standard deviation d_0 by adding thereto a compensation term that is a function of the smallest standard deviation d_0 and said difference between the upper limit *UL* and the mean m_0, to generate the compensated smallest standard deviation $\tilde{d}_0$;

otherwise, no saturation has occurred, wherein $\tilde{d}_0 = d_0$.

22. (Previously presented) The system of claim 21, wherein:

if the mean $m_0 < M$ and the smallest standard deviation $d_0 > m_0 - LL$, indicating saturation has occurred at the lower limit *LL*, then d_0 is compensated as $\tilde{d}_0 = d_0 + K \cdot (d_0 - (m_0 - LL))$, such that K is a compensation factor;

else if the mean $m_0 \geq M$ and the smallest standard deviation $d_0 > UL - m_0$, indicating saturation has occurred at the upper limit *UL*, then d_0 is compensated as $\tilde{d}_0 = d_0 + K \cdot (d_0 - (UL - m_0))$;

otherwise, no saturation has occurred, wherein $\tilde{d}_0 = d_0$.

23. (Previously presented) The system of claim 22, wherein *LL* = 0, *UL* = 255, and *M* = 128.

24. (Previously presented) The system of claim 19, wherein block standard deviations d_n are selected for averaging if $| d_n - \tilde{d}_0 | < \max(\tilde{d}_0, 1)$.